SPRITE CHASER

No. 6

Official Newsletter of the #1 ADAM USERS' GROUP

SPRITE CHASER STAFF

PRESIDENT - Jay H. Forman   EDITOR - Steve George
VICE-PRESIDENT - Steve George   RESEARCH EDITOR - Larry Iocco
SECRETARY - Michele Lovenstein   SOFTWARE LIBRARIAN - Steve George
TREASURER - George Rhodes

President's Message:

I like to start out this newsletter by welcoming our new and I hope will be regular contributors. We hope you have noticed and like our our expanded newsletter. We will try and have a new format for the next newsletter. Write us and tell us about it (pro's & con's). PS: we hope you include an article with you comments. Only you can make our group grow. The Sprite Chaser was late in coming out, for this I apology. We are on the right track and future copies I pray will be on time. Instead of wasting a lot of words, I'll let you get started reading our informative newsletter. Thank you.

WHAT'S NEW

Much has happened since the last newsletter! Please bear with us, I know you all are wondering "What happened to my newsletter???". I think we will finally have the tardiness solved. A major problem with writing this newsletter is I just do not have the time to run the PD exchange, answer the phones, review new products, go to meetings and still hold a full time job! I have found two major contributors (and I would like to thank them now) Gary Daro and Don Zimmerman. For those that have other newsletters please forgive the repeating of information. Gary has some great ideas and will be getting into depth with

Page 1

some of the harder concepts (machine language,BasicPokes&Peeks,etc.) There has been ma
new developments and here are some of them... Eve has put out their 5.25" drives.. Th
are available single sided and double sided. I have not seen one though, and NIAD (an AD
group in Indiana) stated that they have an"over heating problem", which will have be
fixed by the time you read this. Eve has also raised prices by $60. Single-sided drive
$290, fully compatible with the Coleco disk drive. The double-sided drive is 320K and 27
in CP/M... That costs $319. If you already have an ADAM drive you can send yours in fo.
the 2-sided drive for $140.. You need to send also your Disk Manager and CP/M disk for
modification. Unfortunately, you cannot "mix" drives in CP/M. SmartBasic2.0 finally makes
it to the Public Domain! PD disk #37 will be a single disk with 10K of docs on it
(printed out in this issue) PLUS Basic 2.0. A word of caution though: This is a program
that did not go through the usual polishing release software gets. It is NOT compatible
with BASIC 1.0, but can be used in conjunction with it. I would say it is a hobbyists
program useful for experimentation. It does access the 64K expander and does use built- in
sprites! I have also received something called MacAdam which looks to be a pretty full
blown cart-based MacroAssembler for use with the ADAM. It includes an editor and compiler
(but no docs). It looks like it was written by someone in Canada & I am currently trying
to find out if it is available commercially. If it is not and is another Coleco
"deadware" product, then it should find its way into the PD. Look for a massive catalog
from MW Ruth Co. They have been working hard to collate a "one place only" - type of
software and hard ware for the ADAM... That way you can "one-stop shop" and get the latest
in ADAM hardware and software.

INTERESTING PATCHES - D. Zimmerman

Here are two interesting and useful disk patches that were published by various
newsletters to help ADAM users get more out of their systems. The numbers shown will be in
decimal notation with their hex equivalents shown in parentheses followed by a dollar
sign. Any of the disk editor programs available will do the job. 1.) From JKL in
I.E.A.U.G. This patch to any CP/M sysgened disk will increase the usable space of a 64K
memory expander from 55K to 62K. In block 1(1$) change byte number 970 (3CA$) from 55
(37$) to 61 (3D$) and byte number 976(3D0$) from 8 (08$) to 0 (00$). In block number 3
(3$) change byte number 1017(3F9$) from 56 (38$) to 62 (3E$).2.) From P.S.A.N. This patch
will eliminate the famous data bump bug that causes SmartBasic to insert extra spaces in
the DATA & REM statements when you save a Basic program to disk/tape. We all know how
these extra spaces will add up. Do this to a copy of your SmartBasic disk/tape. The only
restriction I can see with this patch is the requirement that your SmartBasic program
start in block number two. Unless someone has REALLY been hacking the heck out of their
SmartBasic tape it should always start in block number two. There are four bytes to change
in block number 17 (11$). Change byte number 208(D0$) from 217 (D9$) to 216 (D8$), byte
number 214 (D6$) from 19 (13$) to 8(08$), byte number 215 (D7$) from 8 (08$) to 55 (37$),
and finally change byte number 216 (D8$) from 55 (37$) to 19 (13$). These patches do work
and all us ADAM owners would like to thank these fine people for sharing their work with
others and encourage them to continue to do so. Thanks... NOTE: Those who have a copy of
SmartBasic 2 can simply LOAD any Basic program into memory and then SAVE it back to disk
and ALL existing extra REM and DATA statement spaces will be removed from the program.
This is an easy way to remove all the extra spaces from any of your Basic programs.

USING SmartBASIC 2.0

Basic 2.0 differs considerably from its predecessors. Many if not all of Basic 1's faults
have been identified and eliminated, and several new features have been added.

The aggravating "growing" DATA and REM LINE Bug has been fixed! And in fact old BASIC 1
files with sloppy DATA and REM lines can be instantly cleaned up by Booting Basic 2.0 and
then LOADing and SAVEing the old file. The utility of sequential Tape or Disc Files has
been considerably improved by fixing the "Backup" Bug. A Backup Copy of saved files is NO
LONGER MADE after APPENDing them. All that's necessary is that the APPENDED FILE be the
LAST PROGRAM on the Tape or Disc. Then there will be no more rapidly filled up tapes with

useless copies, and the SEQUENTIAL FILE feature of Basic can be used directly without resorting to Machine Language routines to overcome BASIC 1's Limitations. Also, DELETEing Programs in Basic 2.0 now actually FREES that space on the Tape or Disc for OTHER PROGRAMS. To be reliable the Programs should be DELETEd beginning with the LAST PROGRAM SAVED and then inward toward the FIRST Program. To DELETE all programs on a tape such as either BASIC tape that can't otherwise be INITiated, just be sure that the LAST PROGRAM DELETED is the FIRST PROGRAM on the Tape or Disc. That will result in a virtually INITiated Basic Tape or Disc with the Full Program Space Available for new programs.

The Line Buffer has been enlarged to now allow a line length up to 255 Characters for direct compatibility with MicroSoft Basic. Line Editing is easier too. The Insert and Delete Keys now work in Basic 2 as well as SmartWriter and the Clear Key Erases the Entire Line. There is another change that really IS an improvement, but takes some getting used to. When editing previously written lines if you should make an Error, the ERROR is ACCEPTED and the Original Line is LOST. This line is then printed on the screen with the "Sad Face" character preceding it. (This Sad Face as well as the Syntax Error "^" and run time Error Messages are always accompanied by the Bell Sound.) It must then be re-written. Sad Face lines will stay in the listing until changed, and may even be SAVEd with the program, but will Break the program when Run, if between sequential lines. This differs considerably from Basic 1.0 where any errors in editing were REJECTED and the original line was returned UNCHANGED. It's good insurance to copy lines before editing them. You'll then have an unchanged line to fall back on if you make an error and lose the line. There is virtually NO imposed POKE LIMIT in Basic 2.0, you can poke up to FEBF (65215). POKEs are rejected above that limit only to prevent Poking into the DCB's which could be catastrophic. Basic 1 Programs with the old POKES TO 16149 and 16150, used to re-set it's Poke Limit SHOULD NOT BE USED with Basic 2.0. These pokes are in a different part of the Basic 2.0 machine language and will blow-up the Basic Boot and require ·re-booting. This damage may not be immediately apparent on all functions and I suspect this is one of the reasons for the reports that Basic 2.0 seems "incomplete". This is true with the old Basic 1.0 COLOR POKES as well. DON'T use a program written for Basic 1.0 that changes the Screen Colors. These programs MUST be modified first.

The NEW COLOR POKES ARE:
POKE 17184 <0-15> for the Border
POKE 17240 <Normal Text *16 + Screen>
POKE 17251 <Inverse Text*16 + Inverse Background>

I have written a 6 Block program for the Public Domain called 1-2 Color that may be used unmodified with either Basic to set all the Basic Text-Screen Color Pokes. It identifies which Basic is booted and pokes the correct addresses.

The Basic 1.0 Graphic Color Pokes SHOULD NOT be used with Basic 2.0 as well. I have not identified the New Graphic Pokes yet. Perhaps someone else has this information.

A small but ·significant improvement (as it seems to point up the level of effort put into the revision) can be seen by running 1-2 Color in both Basics. You'll notice the cursor is visible in Basic 1, even though not necessary since no normal keyboard inputs are required. In Basic 2.0 the cursor is transparent unless waiting for input. The program could have poked a transparent character for the cursor in Basic 1, but if the program was aborted with Control C, without restoring the cursor, it would remain invisible. All this foolishness is unnecessary with Basic 2.0. AND ONCE AGAIN ANY Program that POKES 16953 to alter the cursor in Basic 1.0 MUST NOT BE RUN Before Deleting These Pokes!

The Printer will now start to Print from the Top of the Screen by Pressing the PRINT key on the Keyboard. It can also be STOPPED EARLY by Pressing CONTROL and C together. The Printer now Prints any INVERSE CHARACTERS in the text that have normal characters on the Daisy Wheel, as NORMAL CHARACTERS. Also the Printer can be made to print bi-directional,

as with Basic 1.0, with CHR$(15). This feature was useless however on Basic 1.0 since the printer would lock up and continue printing the same 80 characters forever. This bug has been fixed in Basic 2.0 and the Printer will stop and return control to the program. There is also a NEW SCREEN PRINTING FUNCTION that requires a POKE. To display ANY of the 256 characters including "control characters" on the, screen in the TEXT Mode, at the current cursor position, POKE its ASCII number into 16771 then CALL 16770. The following Line will show all. 5 TEXT: FOR i=0 TO 255: POKE 16771,i: CALL 16770: PRINT " ";: NEXT.

The Basic 2.0 Source File or Program on the Disc/Tape takes up 49 BLOCKS. SmartBASIC 1.0 only required 28 Blocks. ALL this extra code includes a revision of the EOS. This NEW EOS Revision-7 is Loaded into RAM instead of the ROM Revision-5 resident in ADAM that was used with Basic 1.0. This is where the New File Handling Routines come from. Fortunately for the Pokers amongst us the EOS JUMP TABLE is at the same address in RAM and even most of the new routines are simply displaced by a few blocks. Caution is in order however. It's wise to investigate the area of ANY POKE used in your favorite Basic 1.0 Program before attempting to RUN it!

It is also worth noting that even with all this extra source code, Basic 2.0 requires 447 bytes (1/2 Block) LESS memory so that you have additional room for Basic or Machine Code programs even when in STDMEM. Basic 2.0 PRINT FRE(0) yields 26401 as opposed to Basic 1.0's 25954. (Each following NEW). HIMEM default is the same at 53632 but the extra space is revealed by Basic 2.0's default LOMEM of 26960 as opposed to Basic 1.0's 27407. Note too that now BOTH HIMEM AND LOMEM can be set.

Even Block ZERO didn't escape revision. This is the "Boot-STRAP LOADER" and now includes a test for the presence of the 64K Expansion Board as well as A Memory Test of it. Also code to reset the Default Drive to be the Current Drive (the Drive that you're Booting Basic 2.0 from). This means that unspecified Reads or Writes will be to IT and that HELLO will Self-RUN as Expected.

Basic 2.0 correctly INITilizes Discs with 160 Blocks. BLOCKS REMAINING now re-calculated each time CATALOG is requested, from the total of Those Programs NOT DELETED.

NEW COMMANDS

MERGE This command is used like LOAD but does not NEW the program space, so that favorite routines may be stored separately and MERGEd with any existing program. The MERGEd lines will replace any existing lines numbered the same, or will simply be added to the program if numbered differently.

EXTMEM This command accesses the 64K expansion board for additional basic program space. NOTE that when EXTMEM is entered the BASIC 2.0 Tape or Disc MUST BE IN THE CURRENT DRIVE. The Tape or Disc is then Accessed ONLY if the 64K Expansion was tested as Present when Basic 2.0 was Booted. If this is so the Screen will Blank while the Memory Map is reconfigured, then the Title "Coleco SmartBASIC 2.0" will reappear.

PRINT FRE(0) will confirm that your new Expanded Workspace is 90646. (90656 if proceeded by NEW). HIMEM And LOMEM can now be set anywhere in this new workspace but be careful with the values. Reports are that there is no Error Trapping on LOMEM and HIMEM values while in EXTMEM AND AN ILLEGAL ENTRY WILL CRASH BASIC. Note too that one drawback that comes with all this extra room is slower program execution.

STDMEM This command returns you to the normal Basic Map. Once Again the BASIC 2.0 TAPE OR DISC MUST BE IN THE CURRENT DRIVE WHEN STDMEM IS INITIATED. The Tape/Disc is Accessed Again and the Basic Map is configured as "normal". Anything in the Workspace is LOST. PRINT FRE(0) reports 26391 (26401 if proceeded by NEW). (It might prove simpler just to ReBoot the tape or disc from the start)

COMMANDS PDL(1) and PDL(2) in BASIC 2.0 have been INTERCHANGED as referenced to BASIC 1.0. PDL(2) now yields the VERTICAL POSITION for Controller #1 (0-256) and PDL(1) now yields the HORIZONTAL POSITION for Controller #2. ALSO PDL(15) For the #1 controller Or PDL(14) For the #2 Controller now returns an Update on the Position of the SPINNER, used

in the SUPER ACTION CONTROLLER.

BASIC 2.0 SPRITES

Sprites as implemented in Basic 2.0, are really SHAPES each consisting of 32 Bytes of Bit-Maped Memory and can be put anywhere on the screen in ANY of the Display Modes and EACH in any of the 15 HCOLORS! EACH sprite is defined on its OWN SCREEN PLANE, easily visualized as 32 layers of transparencies. That's why each can have it's own color AND that they CAN overlap. The Basic 2.0 Source tape/disc comes with Sprite 1 and 2 already defined. Their 64 bytes are stored in page 0 at 00C0 (192) thru 00FF. Sprites that you define may be stored here or anywhere else in free memory space like any other machine code. Poke 16786 (Low Byte) and 16787 (Hi Byte) to tell the sprite routine where YOUR Sprite Table begins. Before you can use the Default Sprites or any others you must also poke 16788 with a 1. This sets the Sprite Flag and tells the routine that Sprites HAVE been defined.

In "normal" Graphics, Basic 2.0 XDRAWS with Transparent to assume the Background Color (instead of simply Black in Basic 1.0). The Sprite routine uses THIS XDRAW to AUTOMATICALLY Erase That particular Sprite if you Draw it somewhere else on the screen. Needless to say this makes for extremely simple animation routines. To see the two Sprites provided enter this line. 5 TEXT: POKE 16788,1: HCOLOR = 3: DRAW 1 AT 120,70: DRAW 2 AT 120,120. I'm sure they'll be a pleasant surprise.
For more surprises try the following short program.

```
4   REM BASIC 2.0 SPRITE DEMO
5   TEXT:POKE 16788, 1
10  VTAB 3:HTAB 5: PRINT "Remember Text Can ALSO"
15  HTAB 8: PRINT "Be On The Screen"
20  a=4: b=236: c=1: d=1
25  HCOLOR =3:FOR i=a TO b STEP c:DRAW d AT i, 75:GOSUB 35:NEXT:IF
     b=4THEN 40 30 a=236: b=4: c=-1: d=2:GOTO 25
35  FOR t=100 TO 0 STEP -1:NEXT: e=RND(-e):DEF FN f(e)=INT(1+e*RND(1)):
     HCOLOR = FN f(15):RETURN
40  VTAB 6:HTAB 4: PRINT "Wanna See It Again?  y/n ":GET a$
45  IF a$="y" OR a$="Y" THEN VTAB 6: PRINT:GOTO 20
```

After you've tried the above program then try the following one. This Program replaces the existing 2 sprites with 2 new ones. The original ones are erased so if you want to use them again you'll have to re-boot Basic 2.0

```
3 REM NEW SPRITES DEMO FOR BASIC 2.0 ONLY
6 DATA 139,217,169,169,137,137,139,0,139,137,137,249,137,137,139,0,165,7,41,49,41,37,165,
    0,161,33,33,33,33,33,189,0
7 DATA 209,0,0,226,0,0,229,0,47,40,72,143,72,40,47,0,85,0,0,170,0,0,85,0,120,68,68,120,
    80,72, 68,0
8 FOR x=192 TO 255:READ d:POKE x, d:NEXT
9 TEXT:POKE 16788, 1
10 VTAB 3:HTAB 5: PRINT "Remember Text Can ALSO"
15 HTAB 8: PRINT "Be On The Screen"
20 a=4: b=220: c=1
25 HCOLOR =3:FOR i=a TO b STEP c:DRAW 1 AT i, 75:DRAW 2 AT i+16, 75:   GOSUB 35:NEXT:IF
    b=4 THEN 40
30 a=220: b=4: c=-1:GOTO 25
35 FOR t=100 TO 0 STEP -1:NEXT:RETURN
40 VTAB 6:HTAB 4: PRINT "Wanna See It Again?  y/n ":GET a$
45 IF a$="y" OR a$="Y" THEN VTAB 6: PRINT :GOTO 20
```

MAKING YOUR OWN SPRITES

If you have SmartLOGO then you can very quickly define your own sprites using the Logo SHAPE EDITOR. LOGO Shapes are defined in exactly the same way as SmartBasic 2.0 Sprites. This is described fully in Chapter 6 of the Logo Reference Manual. The simple procedure

is: Boot SmartLogo Program and answer NO to the Demos. Then Enter after the Logo Prompt ES 32 (RTN). LogoShape 32 is normally Blank so an empty 256-Block Box 16 blocks High by 16 Blocks wide will fill the screen. (The Turtle Shape, actually another Sprite will be visible in the center of the screen but will not affect the following procedure. If it's objectionable, it can be HIDDEN first following instructions detailed in the LogoManual)

To FILL a Block move the LogoCursor to that block and Press HOME. Pressing Home when the cursor is already in a FILLed Block will ERASE that Block and return it to Transparent. Remember that even though Black is used to FILL blocks in the LogoEditor these may be later defined as any Color. If you ran program #2 above then you saw that Sprites Can Also Be configured to look like Text. These Text Characters should each be 7 blocks Hi by 5 or less Blocks wide with a space underneath and to the right. This will allow a sprite to contain 2 lines by 3 characters each. For 2 Lines of Text several sprites can be defined and placed side by side.

After you have your first Sprite Defined the way you want it Press SMARTKEY [VI]. This will Save its Shape in the Workspace (Its not really necessary to save it on the tape). The Editor will disappear and when the LogoCursor returns Enter MAKE "SPRITE GETSH 32 (RTN). And then Enter PR :SPRITE (RTN). The 32 decimal numbers that define a Sprite will be listed in their correct order. Just copy them onto a sheet of paper. If you want to define more sprites you can use Shape 32 over or any other Of the 60 LogoShapes. Any changes that you make will not be permanent unless you follow the procedure to make them so detailed in the Logo Instructions. After you have the list of 32 numbers you can copy these into Program #2 above in place of one of the DATA Lines. When you run the program your Sprite will be displayed.

BIT-MAPPING YOUR OWN SPRITES

If you don't have the Logo Program or if you just want to learn how to define your own sprites by plotting a bit-map try the following simple procedure.

On a sheet of graph paper enclose a 256 block square, 16 Blocks high By 16 Blocks wide. Starting at the TOP RIGHT corner, outside the square along the top line, from Right to Left write above each column: 1,2,4,8,16,32, 64,128,1,2,4,8,16,32,64,128. Now starting at the Top Left Corner, outside the square from Top to Bottom, Number each of the 16 Rows 1 thru 16. Now on the RIGHT Side, outside the square from Top To Bottom, number these same rows 17 thru 32. Finally, in the Center of the Square Draw a Vertical Line from Top To Bottom, starting between the numbers 1 and 128 along the Top.

What you now have is a graphic representation of the 32 bytes in memory that define a sprite arranged as they are displayed on the screen as two side by side 16 Byte columns. Each 1/2 line of 8 blocks is an 8 bit Byte. Blocks that you FILL assume the value of 1 and empty Blocks are 0. To easily convert these Binary Bytes to the decimal numbers required by Basic just look up at the top row of numbers. If a Block is FILLED, ADD This Number to the Total for this 1/2 line Byte (Numbered 1-32). Empty 1/2 lines are 0. Write these 32 numbers down in order and with a comma between each one and copy them into either of the Data Lines in the 2nd Program above inplace of the existing numbers. When you RUN the program you'll then see your own sprites displayed.

SMART BASIC V2.0 COMMAND LIST

ABS	DIM	IF	NOTRACE		
AND	DRAW	IN#	ON		
APPEND	END	INIT	ONERR		
ASC	ERRNUM	INPUT	OPEN		
AT	EXP	INT	OR		
TN	EXTMEM	INVERSE	PDL	RETURN	STORE
BLOAD	FLASH	LEFT$	PEEK	RIGHT$	STR$
BRUN	FN	LEN	PLOT	RND	TAB

BSAVE	FOR	LET	POKE	ROT	TAN
CALL	FRE	LIST	POP	RUN	TEXT
CATALOG	GET	LOAD	POS	SAVE	THEN
CHR$	GOSUB	LOCK	POSITION	SCALE	TO
CLEAR	GOTO	LOG	PRINT	SCRN	TRACE
CLOSE	GR	LOMEM	PR#	SGN	UNLOCK
CLRERR	HCOLOR	MERGE	READ	SHLOAD	USR
COLOR	HGR2	MID$	RECALL	SIN	VAL
CONT	HIMEM	MON	RECOVER	SPC	VLIN
COS	HLIN	NEW	REM	SPEED	VPOS
DATA	HOME	NEXT	RENAME	SQR	VTAB
DEF	HPLOT	NOMON	RESTORE	STDMEM	WAIT
DEL	HTAB	NORMAL	RESUME	STEP	WRITE
DELETE		NOT		STOP	XDRAW

Adam Software and ROM-cartridge List

As you know there are many carts out there and this is the start to a REAL comprehensive listing of those available. I have seen many of the ones below that this author has not seen and we are currently researching if they can be put in the Public Domain. This list will also include some Adam software never released also. The ones I have seen and currently have a copy of are:

Cabbage Patch 128K (Has better graphics than cart version) * Fall Guy (uses driving module) by 20th Century Fox * Yokes On You by 20th Century Fox * MacAdam..(a cart based Macro Assembler) * Speed Checker..(checks the speed of a DDP drive) * Adam Demo..A looping demo of the capabilities of the ADAM Adam * Test Cart..Designed a a simple way to test system peripherals, memory, etc.

Basic2.0..Uses extended memory, built in sprites 5 screen DK jr...A version with an extra screen built in Market Monitor..A program released in Canada that uses the modem to access a"Dow Jones" type of retrieval service uses modem in BASIC!

Rumors:

There is a TON of software rumored to have been developed and is "out there". Some of them (and we'll verify these as they turn up):Burgertime disk version AdamlinkIII..1200Baud version Carts: Cranston Manor, Mr.Turtle, Skiing, HorseRacing, Chess Challenger, RipCord, Spectar, Tunnels&Trolls, Return of the Jedi, DinoEggs

VIDEO GAMES PRODUCED ON CARTRIDGE FOR COLECOVISION/ADAM SYSTEM

This is meant to be a comprehensive listing. This list is correct to the best of my knowledge. If you have corrections, additions or can contribute some of the missing information please send it to the editor of this newsletter.
Thank you. Dave Palmater

<<<CREATED BY: company responsible for the writing of the game. The few exceptions are when the concept was licensed from another and developed as a video game (e.g. War Games)
<<<ISSUED BY: company who actually released implementation of game for COLECOVISION/ADAM.
<<<Copyright dates are supplied when available.
<<<SIZE (given in K): calculated when game is copied to Disk/DP using CopyCart+ (MMSG). Games saved as CP/M files are approximately 1K longer due to loader that is saved with ea.

NAME OF GAME	CREATED BY	ISSUED BY	SIZE	NOTES
A.E.	Broderbund	Coleco	???	13
Alphabet Zoo	Spinnaker (1983)	Spinnaker (1984)	16K	
*Amazing Bumpman	???	CBS	???	NS
Antarctic Adventure	Konami (1984)	Coleco (1984)	16K	

Aquattack	Interphase	Interphase (1984)	16K	20
Artillery Duel	Xonox	Xonox (1983)	16K	20
Baseball	Coleco	Coleco (1983)	24K	3
B.C.'s Quest	Sydney	Sierra (1983)	16K	22
Beamrider	Cheshire	Activision (1984)	16K	
Blockade Runner	???	???	???	NS
Boulder Dash	First Star	Microsoft (1984)	16K	
Brain Strainers	???	Coleco	???	NS
Buck Rogers Planet Zoom	Sega	Coleco (1983)	24K	1
Bump N Jump	???	???	20K	8
Burgertime	Data East (1982)	Coleco (1984)	16K	
Cabbage Patch Kids Park	???	Coleco	???	NS
Cabbage Patch Pic. Show	???	Coleco	???	NS
Campaign '84	Sunrise	Sunrise (83)	16K	10
Carnival	Sega	Coleco (1983)	16K	
Centipede	Atari	Atari (1983)	15K	
Chopperlift	Broderbund	Coleco	???	23
Chuck Norris Karate	Xonox	Xonox (1983)	16K	
Congo Bongo	Sega	Coleco (1984)	24K	
Cosmic Avenger	Universal	Coleco (1983)	16K	
Dambusters-	Sydney	Coleco (1984)	32K	2
Decathalon	Activision (83)	Activision (84)	16K	9,15
Defender	William (1980)	Atari (1983)	24K	20
Destructor	Coleco	Coleco (1984)	32K	7
Donkey Kong	Nintendo	Coleco (1982)	16K	1
Donkey Kong Jr.	Nintendo (1982)	Coleco (1983)	16K	1
Dragonfire	Sherman (1984)	Coleco (1984)	16K	
Dukes of Hazzard	Warner	Coleco (1984)	32K	7
Dr. Seuss Mix-Up Puzzle	Coleco	Coleco (1984)	16K	
Evolution	Sydney	Sydney (1983)	16K	
Facemaker	Spinnaker (1983)	Spinnaker (1984)	16K	
Fall Guy	???	???	???	7, NS
Fathom	Imagic (1983)	Coleco (1983)	16K	
Flipper Slipper	Spectravideo	Spectravideo	16K	
Football	Coleco	Coleco (1984)	32K	3
Fortune Builder	Circuits/Systems	Coleco (1984)	32K	
Fraction Fever	???	Spinnaker	???	NS
Frantic Freddy	Spectravideo	Spectravideo	16K	
Frenzy	Stern (1982)	Coleco (1983)	24K	
Frogger	Sega	Parker Bros (1983)	16K	
Frogger 2	???	???	16K	
Frontline	Taito	Coleco (1983)	24K	2, 3
Galaxian	Atari	Atari (1983)	20K	
Gateway to Apshai-	Epyx	Epyx (1984)	16K	19
Grog's Revenge	Sydney (1984)	Coleco (1984)	24K	22
Gorf	Bally (1981)	Coleco (1983)	16K	
Gust Buster	Sunrise	Sunrise (1983)	16K	10
Gyruss	Parker Bros.	Parker (1984)	16K	
Heist, The •▪	Livesay	Microfun (1984)	24K	21
Hero	Van Ryzin	Activision (1984)	16K	
Illusions	Nice Ideas ('84)	Coleco (1984)	16K	
James Bond	Parker Bros.	Parker (1984)	16K	11
Juke Box	Hankanson	Spinnaker (1984)	16K	
Jumpman Jr.	Epyx	Epyx (1984)	16K	
Jungle Hunt	Taito (1982)	Atari (1983)	24K	
Ken Uston's BJ-Poker	Coleco	Coleco (1983)	16K	
Keystone Kapers	Kitchen (1983)	Activision (1984)	16K	

Lady Bug	Universal	Coleco (1982)	16K	
Lancelot	Xonox	Xonox (1983)	16K	
Learning with Leeper	Monier	Sierra (1983)	16K	
Linking Logic	SCC	Fisher-Price (84)	16K	
Logic Levels	???	???	???	NS
Looping	Venture Line	Coleco (1983)	16K	
Memory Manor	???	???	???	NS
Miner 2049'er	Livesay	Microfun (1983)	24K	
Monkey Academy	???	???	???	NS
Montezuma's Revenge	Parker Bros.	Parker Bros.	12K	24
Moonsweeper	Imagic (1983)	Coleco (1983)	16K	
Motocross Racer	Xonox	Xonox (1984)	16K	
Mountain King	E.F.Dyer (1983)	Sunrise (1984)	16K	10
Mouse Trap	Exidy	Coleco (1983)	16K	
Mr. Do	Universal	Coleco (1983)	24K	
Mr. Do's Castle	???	???	???	NS
Nova Blast	???	???	???	NS
Oil's Well	???	Sierra	???	NS
Omega Race	Bally (1981)	Coleco (1983)	16K	6
One on One	???	???	24K	14
Past Finder	???	???	???	NS
Pepper II	Exidy	Coleco (1983)	16K	
Pitfall	Crane	Activision (1983)	16K	
Pitfall II	Crane	Activision (1983)	16K	
Pitstop	Epyx	Epyx (1984)	16K	8
Popeye	Parker Bros.	Parker Bros (1983)	16K	
Q*bert	Gottlieb (1983)	Parker Bros.	8K	
Q*bert's Qubes	Gottlieb	Parker Bros.		
Quest for Quintana Roo	Sunrise	Sunrise (1984)	16K	10
River Raid	Activision ('82)	Activision ('84)	16K	
Robin Hood	Xonox	Xonox (1984)	16K	
Rock 'n' Roll	K-Tel (U.K.)	K-Tel (U.K.)('84)	16K	17
Rocky Super Act Boxing	Coleco	Coleco (1984)	20K	3
Roc 'n Rope	Konami (1983)	Coleco (1984)	24K	
Rolloverture	???	Sunrise (1983)	16K	10
Sammy Lightfoot	Sierra On-Line	Sierra (1983)	16K	
Sector Alpha	???	???	???	NS
Sewer Sam	Interphase	Interphase (1984)	24K	20
Skiing	???	CBS	???	NS
Slither	Century II	Coleco (1983)	16K	5
Slurpy	Emag	Emag (1984)	14K	
Smurf Rescue	???	Coleco	???	NS
Smurf Paint & Play	???	Coleco	???	NS
Soccer	???	CBS	???	3, NS
Space Fury	Sega	Coleco (1983)	16K	
Space Panic	Universal	Coleco (1983)	16K	
Spectron	Spectravideo	Spectravideo (83)	16K	
Spy Hunter	Bally (1983)	Coleco (1984)	32K	4
Squish 'em Sam	???	???	???	NS
Star Trek	Sega	Coleco (1984)	24K	4
Star Wars	Atari	Parker (1984)	16K	12
Subroc	Coleco	Coleco (1983)	20K	2
Super Cobra	???	???	9K	9
Super Cross Force	Spectravideo	Spectravideo	16K	
Tapper (Root Beer)	Bally (1984)	Coleco (1984)	32K	
Tarzan	Coleco	Coleco (1982)	24K	18
Telly Turtle	Carousel (1983)	Coleco (1984)	???	NS

Threshold	Sierra	Sierra (1983)	16K	
Time Pilot	Konami	Coleco (1983)	16K	
Tomarc the Barbarian	Xonox	???	???	NS
Tournament Tennis	D + L (1984)	Coleco (1984)	16K	16
Turbo	???	???	???	7, NS
Tutankhamen ➤	???	Parker Bros.	16K	9
Up N' Down	Sega	Sega (1984)	16K	
Venture	Exidy	Coleco (1982)	16K	
Victory ♦	Exidy	Coleco (1983)	20K	
5 Video Hustler (Pool)	Konami	Konami (1984)	8K	
War Games '	U.A. 1983	Coleco (1984)	24K	
War Room	Harris	NAP	24K	
Wing War	Imagic (1983)	Imagic (1983)	16K	
Wizmath	Sierra			
Word Feud	???	???	???	
NS Yolks On You	???	???	???	NS
Zaxxon	Sega	Coleco (1982)	24K	1
Zenji	Hubbard	Activision (1984)	16K	
2010 - The Graphic Adv.	Coleco	Coleco (1984)	32K	13

<<<NOTES>>>
1 Super Game also produced
2 Super Game developed but not released
3 Requires Super Action Controllers
4 Can utilize Super Action Controllers but does not require them
5 Requires Roller Controller
6, Can utilize Roller Controller but does not require it
7 Requires Driver Action Controller
8 Can utilize Driver Action Controller but does not require it
9 Unsure of correct name - no title screen
10 Same game issued by Sunrise on DP/Disk
11 Subtitled Diamonds are Forever
12 Concept and name licensed from Lucas Films
13 2010 - The Text Adventure Game was also released
14 Seems to be an implementation of Dr. J./Larry Bird One on One
15 Seems similar to Summer Games
16 D + L = Imagic
17 Full title - It's Only Rock 'n' Roll
18 Full title - From Out of the Jungle...Tarzan
19 Temple of Apshi Super Game Created [but not released (?)] by Expy
(1984) 20 Controls on this game do not work properly (bad cart?)
21 I can't get the Thief to do anything but run off the screen and not
   come back. Something I don't know or do I have a bad version?
22 Released on DP/Disk as a part of The Best of B.C.
23 Released on DP/Disk as a part of The Best of Broderbund
24 Will not run using Super Action controllers
<<<NS Not seen by me>>>

<<<ADDENDA>>>
Developed by Coleco for sale but never marketed: Consumer Diagnostic Checkout (Systems test with graphics), Chess Challenge * Developed by Coleco for in-store use: ADAM Demo (demonstrated ADAM and Software Packages including some never marketed) * Developed by Coleco for in-house use: Tape Utility 2 (Copy program with verify and compare feature), Basic Restorer (Restores Smart Basic Tapes)

Print Fix Again
Some ADAMs may have a problem printing out text exactly as the screen shows it if the left

hand horizontal margin is set to one. This bug appears in the line following a line that has only a carriage return in IF the text on the following line BOTH starts in column number one AND if the word starting in column one is underlined. All of the following conditions must be present AT THE SAME TIME for this bug to show up: 1) Left hand horizontal margin set to one. 2) The line PRECEDING the troubled line contains a blank line (only a lone carriage return) with NO text. 3) The text in the line following the blank line BOTH starts in column number one AND is underlined. When the ADAM printer encounters a lone carriage return it apparently issues the infamous signal for the one and one half line feeds to the printer. If the text in the following line starts in column number one the print head must return to the left until it contacts the switch that tells it that it is now at the left most side of the roller. ADAM seems to have a problem backing up to underline text if the text starts in column number one IF the left margin is set to one. It appears that ADAM needs to back up one space before the underlined text to do the underlining. This problem must have something to do with the one and one half line feeds because if there is ANYTHING on the above line with the carriage return this bug does not show up. The head can not back up from column number one because ADAM uses the contact switch to place the print head at the left most position and this position is where column number one will be printed. I do not know if all ADAMs have this bug but both our ADAMs do it. They both are almost two and one half years old so perhaps the newer ones have been fixed. FIXES: Any one of these will correct this bug. 1)Don't set the left hand margin to one. Any other setting will not have this bug. 2)If the left hand column is set to one do not start any underlined text following a blank line in column number one. Start the underlined text in column two or greater. 3) If the left hand margin MUST be set to one and underlined text MUST start in column number one following a blank line, THEN use the old "trick" of placing a period in the blank line before entering the carriage return. This period is the smallest character on the print wheel and is not too noticeable. It helps if the period and carriage return is placed out at the right hand margin setting. This also could be done to eliminate the extra one half line feeds for blank lines. If the extra carriage returns bother you so much on the ADAM printer simply take a razor blade and trim off one of the seldom used characters on your print wheel and enter this character before the carriage return for a blank line. The cut off character will not print and you have eliminated the extra one half line feed and the column one bug. It seems like a lot of trouble to go through for such minor bugs.

Fun With MultiCart Backup

How many of you own or use MultiCart Backup by Best Software? How many wish that they could transfer games to and from a disk/DDP? I know I have some disks with repeats on them and I'd rather substitute another game for one that is already on the disk... Well, there is a way to accomplish this... The only problem is one has to be pretty adept with one of the various editors I always talk about... My favorite is JKL Utilities (written by Joel Lagerquist, a former ADAM programmer). It will dump the contents of any block in HEX (short for Hexadecimal, base 16, for more info read last issue) and display them in "pages" of 256 bytes each. They can then be easily modified and written to any of the four ADAM drives. MultiCart Backup is a nice cartridge utility because it is only 2 blocks long, written in assembly language and almost totally fills a disk or tape. It also seems to copy all carts (some of the other cart utilities cannot copy the larger carts because there is not enough room for the copy program and the cart in memory). It also uses block 1 for the directory and it is relatively easy to figure out. What follows is part of an actual directory listing in Hex with ASCII on the right side. 03E8: FFFFFFFFFFFFFF79 ^^^^^^^ y03F0:6F6B65736F6E7520 okesonu03F8: 2000FF0210FF0112 .^.^..^. This is found near the end of block 1 (as denoted by the leftmost numbers - 03F8 is very close to the end of the block) Note: a "block" is 256X4 or 1024 bytes long in decimal (base 10). This disk only has one cart on it called "yokesonu" which stands for a cart called Yoke's On You (by 20th Century Fox-I think never released). You can tell because the title is to the right... The Hex numbers that stand for these letters is in the middle... In the back of the SmartBasic Manual you can find a listing of ASCII/Hex numbers and there equivalents. So small case y=79 Hex. To change the title to something else, just

change the Hex numbers to different letters and write the block back to disk. You have to be real careful, though. The title, once written HAS TO BE THE SAME LENGTH AS THE ONE ALREADY ON THE DISK. If you wanted to change "yokesonu" to a different title, it has to be 8 letters long.MultiCart writes the directory from the end towards the beginning. Yoke's, being the first, is the last one listed in the directory (Confusing, eh??).. If you were to copy another cart to the disk, it would come BEFORE Yoke's. Now for the confusing stuff! The format for the directory is like this:FFFFFF and so on means an empty space (bytes are designated by two letter/number combinations). Next is the actual "filename" in this case Yoke's On You... It can be up to 9 spaces long... A blank is denoted by a 20. After EACH filename MUST come a 2000FF (end of file), then the location of cart in HEX, and the length of the cartridge in blocks. Again from the actual listing:03E8: FFFFFFFFFFFFFF79 ^^^^^^^ y03F0:6F6B65736F6E7520 okesonu 03F8:2000FF0210FF0112 .^.^..^. This follows what we have been saying... After the title, starting on the third line is "2000FF", then 02 (the cart is at block 2), then 1(the cart is 16 blocks long...10H=16 decimal). The last three bytes are also very important as they are part of the directory... "FF" stands for "End of Directory", "01 is how many carts are one the disk/tape (in this case only 1), and "12" is the next available block to copy the next block to. Since the cart we have one there already starts at two, and is 16 blocks long (or 10H), then the next available would have to be "12H".Now once you know where a cart is on a disk, you can then find them easily.. When your editor specifies block number, you then type (in this case) "2"... you will see at address 0000, block 2 the number/letters "55AA" (hex) or "AA55". Usually, carts that begin with 55AA load up with the "Colecovison" title and the others (AA55) go directly to the game without the 12 second titlescreen... (you can experiment WITH BACKUPS and just reverse these two numbers and see what it does).Now, that is all you need to know to move carts around... If I wanted to move Yoke's On You to another disk that already had carts on it, I would follow these steps: (Of course always work with backups!)1.) With your editor, load up block 1 from the destination disk.2.) Go to the end of the block and look for the topmost title (this was the last one written). Count 5 bytes in (or ten spaces-2 spaces=1 byte). That is where the LAST letter of the title will go. Count nine more bytes in (or 18 spaces if you are using a screen-oriented editor like JKL Utilities). Then write the title of the game you are bring over. (It should look like the above example)3.) After the last letter put a 2000FF for end of file. 4.) Then put the block number where it is going 5.)Next put the number of blocks long the cart is. 6.) Finally, go to the last three bytes of block 1 and change the numbers of carts on disk (second to last), and the next available block (last byte).7.) You can then do a block move with your editor (if it has the capability), Put in Source disk, specify where the cart is, then how many blocks you want to transfer (two drives make this easy), then put the destination in. 8.) Lastly, check your work..boot it up.. If the the game plays you have figured it out... If it doesn't (and this has happened to me!), go back to step one, read the directory of the destination, and check it out.. Usually there is a mistake.If this sounds like Greek (or Latin) it is! Only with practice does this come easy. I did not start out just loading blocks in, changing them, then writing them back... First you need to explore and be patient! Some of the cart programs make moving them around REAL easy... CopyCart+ is one of these (although I found it did not always copy the cart especially the long ones).I hope this was relatively easy to follow, but it is pretty hard to describe without "hands-on" experience.. let me know if you have any questions!

HACKER'S GUIDE TO ADAM, VOL. 2 by Ben Hinkle, Review by G. Daro
The second edition of the Hacker's Guide is a complete outline of SmartBASIC. The book explains how ADAM stores and executes BASIC programs & how it responds to keyboard input. This volume is much easier to understand than the previous work from Ben and Peter Hinkle. There is a very clear explanation of how BASIC commands are tokenized and how the 'crunch code' is stored in RAM. Chapter 6 is dedicated to explaining how BASIC commands are interpreted, or parsed, and how errors are generated. The parse routines for commands and math functions are clearly detailed. There are chapters detailing keywords, screen routines, tape routines, graphics and also a chapter dedicated to the data tables used

by BASIC. With the information contained in volume 2, you can easily change or add new BASIC commands. In fact Chapter 11 does just that. Ben Hinkle provides us with a new BASIC command for sound that is very easy to use. For all the graphics lovers, how about Sprite commands in BASIC, something Coleco never gave us. In addition to setting up the sprite, there is a command for testing if two sprites collide! How about a 40 column display in BASIC, that's also in chapter 11. There are some other nifty modifications and fixes like the DATA bump bug that adds those destructive spaces. There are even several schematics of the ADAM and ColecoVision boards. All of the new commands and fixes are contained in a long program printed in the book and also available on DDP, a worthwhile addition. To fully understand and use the document, you should have a basic understanding of the Z80 and how it uses registers and memory locations. The book along with the DDP are available for $17.95 from: M.W. Ruth Co. 510 Rhode Island Ave., Cherry Hill, NJ 08002

Basic Program - Copy Program

```
5REM COPY V1.2 8/17/1986 By G. Daro
10HIMEM :27951:POKE 16953, 32: q=0
20GOSUB 500
30DATA 1,0,0,17,0,0,62,0,33,48,109,205,243,252,201
40FOR x=1056 TO 1070:READ d:POKE x, d:NEXT
50IF sd=dd THEN  m$="MASTER":GOSUB 300
60POKE 1063, sd:POKE 1066, 121:POKE 1068, 243
70FOR i=0 TO 19
80VTAB 5:? "READING BLOCK: "; sr:POKE 1060, sr:CALL 1056
90POKE 1066, PEEK(1066)+4
100IF sr=er THEN  q=i:GOTO 200
110 sr=sr+1:NEXT i
200IF sd=dd THEN  m$="SLAVE":GOSUB 300
210POKE 1063, dd:POKE 1066, 121:POKE 1068, 246
220FOR i=0 TO 19
230VTAB 7:? "WRITING BLOCK: "; sw:POKE 1060, sw:CALL 1056
240POKE 1066, PEEK(1066)+4
250IF sr=er AND q=i THEN POKE 16953, 95:? "END":CLEAR:END
260 sw=sw+1:NEXT i
270GOTO 50
300VTAB 12:? "Place "; :IF m$="MASTER" THEN ? m$; :GOTO 320
310FLASH:? m$; :NORMAL
320? " in drive..."
330? "Hit any key when ready...":GET q$:RETURN
500TEXT:VTAB 5
510? "SELECT DRIVES:"
520VTAB 10
530HTAB 10:? "DISK DRIVE #1: 4"
540HTAB 10:? "DISK DRIVE #2: 5"
550HTAB 10:? "TAPE DRIVE #1: 8"
560HTAB 10:? "TAPE DRIVE #2: 24"
570VTAB 15:INPUT "SOURCE DRIVE ?"; sd
580INPUT "DESTINATION DRIVE ? "; dd
590TEXT
600VTAB 5:INPUT "START READ BLOCK: "; sr
605VTAB 5:? "                         "
610VTAB 5:INPUT "END READ BLOCK: "; er
620VTAB 5:INPUT "START WRITE BLOCK: "; sw
625IF sd=dd THEN 660
630TEXT
635?:?:?:?
640? "PLACE SOURCE MEDIA IN DRIVE "; sd; ", PLACE DESTINATION MEDIA IN DRIVE "; dd; "."
645?:?
```

```
650INPUT "HIT RETURN...."; q$
660TEXT:RETURN
```
<<<Doc file below explains how to use above program>>>
<div align="center">COPY THOSE TAPES AND DISKS!   by Gary Daro</div>
Coleco did not provide us ADAM owners with a method of making backup copies of system software such as SmartBASIC. If something happens to your DDP or disk, you are out of luck! You should have several backup copies of all your important programs and data files. User written programs and data files can be copied by using the Coleco DOS copy utility, but that doesn't help you with the system software.

To copy system software, you need a program that reads the blocks on the DDP or disk into RAM, then writes them to the designated drive, thus producing an exact copy of the source media. There are no BASIC commands to do this, you must use machine code and the ADAM operating system in order to get around the limitations (intentional?) of SmartBASIC. There are several good commercial utilities that have good block copy programs. Both Uncle Ernie's Toolkit and Packcopy provide block copiers as well as other useful utilities. They are inexpensive ways of protecting valuable software.

The following is a short SmartBASIC copy program that uses a machine language routine and the operating system routines to read and write directly to the tape or disk. The explanation provided should help you understand how you can have your cake & eat it too!

The heart of the program is the machine language routine that is poked into an unused area of memory starting at 1056. The program calls an operating system routine that reads one block of data. You must first set up the Z80 registers with certain information. This is done by loading (LD) the register or register pair with the data, then you CALL the OS routine. Here are the Z80 op codes for the program:

```
LD    BC,nn  (1,0,0)  )Sector number of disk or tape to
LD    DE,nn  (17,0,0))read/write,  4 bytes are used.
LD    A,N    (62,0) Drive to use for read or write (can be different).
LD    HL,nn  (33,48,109) Start address of buffer for read and write.
CALL  FCF3   (205,243,252) Call the OS routine for read/write
RET (201) Return to BASIC.
```
Routine reused for write by changing CALL address from FCF3 to FCF6 by poking #'s BASIC.

Line 10:   The HIMEM statement sets up a 20k buffer for the read and write operations. Poke 16953,32 just sets the cursor to a blank for appearance.
Line 20:   The subroutine starting at 500 to 660 gets the source and destination drives (all 4 can be used in any combination).
Line 30:   Data for ml routine.
Line 40:   Poke ml routine into RAM.
Line 50:   If copying to the same drive, then prompt user to change media.
Line 60:   Poke in the source drive, the starting address of the buffer (remember numbers are stored lo byte, hi byte), address of the OS read routine.
Line 70:   Start loop to read 20 blocks.
Line 80:   Display to let user know status of read, poke block to read, call ml routine.
Line 90:   Increment the buffer address by 1024 (4 time high byte).
Line 100: If start read & end read are the same, then the read is over and skip to write.
Line 110: If not, increment block to read (sr=sr+1), loop again.
Line 200: If drives are the same, prompt user to change media.
Line 210: Poke target drive, poke start of buffer, poke address for OS write.
Line 220: Start write loop.
Line 230: Display to let user know status, poke the block to be written, call ml routine.
Line 240: Increment the buffer by 1024.
Line 250: Checks to see if the process is over, if so, clear memory and end.
Line 260: Write next block.

Line 270: Start read procedure again.
Line 300 to 330 prompts for reading and writing to same drive.
I hope you find this program useful. It is intended for you backup copies of software that you have purchased. Please write to me in care of Sprite Chaser if you have any problems or suggestions for improving the program.   Gregory R. Daro

Basic Program-CARDS

```
10REM Adapted/modified to ADAM by D.Zimmerman
4REM from Sept 84,Popular Computing
5POKE 17059, 12:POKE 17115, 31:POKE 17126, 252:POKE 16953, 42:TEXT:HOME
10DIM cv$(14), d(54)
20FOR c=1 TO 14
30READ cv$(c)
40 cv$(c)=LEFT$(cv$(c)+" ", 2)
50NEXT c
60DATA A,2,3,4,5,6,7,8,9,10,J,Q,K,WC
70 bk$="X"
80 nc$=" "
90 wc=14
100 m2$="   "
110 m3$=" - "
113INPUT "ENTER A RANDOM NUMBER "; x
115 x=RND(-ABS(x))
120HOME
130? "CONCENTRATION"
140?
150INPUT "NUMBER OF PLAYERS (1-4) "; px
152?
160IF px<1 OR px>4 THEN GOTO 140
170FOR pn=1 TO px
180 pn$(pn)=""
190? "NAME OF PLAYER "; pn
200INPUT pn$(pn)
210IF pn$(pn)="" THEN  pn$(pn)="PLAYER "+STR$(pn)
220 sc(pn)=0
230NEXT pn
240FOR c=1 TO 54
250 d(c)=0
260NEXT c
270FOR w=1 TO 2
280 fv=14
290 c=INT(RND(1)*54+1)
300IF d(c)>0 THEN GOTO 290
310 d(c)=fv
320NEXT w
330FOR n=1 TO 52
340 c=INT(RND(1)*54+1)
350IF d(c)>0 THEN GOTO 340
360 d(c)=n-INT(n/13)*13+1
370NEXT n
380 cl=54
390 pn=1
400 cn(1)=0
410 cn(2)=0
420FOR c=1 TO 2
430GOSUB 950
440?:?
```

```
450INVERSE:? " "; pn$(pn); " "; :NORMAL:? " SELECT CARD # "; c
452?
460INPUT "ENTER COLUMN NUMBER: "; co
462?
465INPUT "ENTER ROW NUMBER: "; ro
470 ro=INT(ro)
480 co=INT(co)
490IF ro>=1 AND ro<=9 AND co>=1 AND co<=6 THEN GOTO 530
500?:INVERSE:? "ERROR: CHOOSE-> COLUMN=(1-6)      ROW=(1-9) ":NORMAL
510GOSUB 1100
520GOTO 430
530 cr=(ro-1)*6+co
540IF c=2 AND cr=cn(1) THEN GOTO 590
550IF d(cr)>0 THEN GOTO 620
560INVERSE:? " THAT CARD HAS BEEN TAKEN! ":NORMAL
570GOSUB 1100
580GOTO 430
590INVERSE:? " YOU JUST TOOK THAT CARD!":NORMAL
600GOSUB 1100
610GOTO 430
620 cn(c)=cr
630NEXT c
640GOSUB 950
650IF d(cn(1))=d(cn(2)) OR d(cn(1))=wc OR d(cn(2))=wc THEN GOTO 680
660?:?:INVERSE:HTAB 9:? " NO MATCH ":NORMAL
670GOTO 740
680?:?:INVERSE:HTAB 9:? " ** MATCH **":NORMAL
682FOR i=1 TO 10:? CHR$(7); :NEXT i
690 d(cn(1))=0
700 d(cn(2))=0
710 sc(pn)=sc(pn)+2
720 cl=cl-2
730IF cl-2 THEN GOTO 820
740GOSUB 1100
750 pn=pn+1
760IF pn<=px THEN GOTO 810
770 pn=1
780?
790GOSUB 900
800GOSUB 1100
810GOTO 400
820?
830? "NO CARDS LEFT -- GAME OVER!!"
840?
850GOSUB 900
860?
870INPUT "PLAY AGAIN? (y/n) "; yn$
880IF LEFT$(yn$, 1)="y" OR LEFT$(yn$, 1)="Y" THEN GOTO 120
890END
900HTAB 6:INVERSE:? " ** SCORES ARE ":NORMAL
910FOR pl=1 TO px
920? "      "; pn$(pl), sc(pl)
930NEXT pl
940RETURN
950GOSUB 3000
951HOME:HTAB 8:INVERSE:? " CONCENTRATION ":NORMAL:?
952? "     ";
```

```
953FOR i=1 TO 6:INVERSE:? " ";
954? "C"; m2$; :NEXT:NORMAL
955?:? "    ";
956FOR l=1 TO 6:INVERSE:? " ";
957? l; :n2$; :NEXT:NORMAL
958?:?: x=1
959? " "; :INVERSE:? "R"; x; :NORMAL:? "   "; : x=2
960FOR cq=1 TO 54
970IF d(cq)>0 THEN GOTO 1000
980INVERSE:? nc$; :NORMAL:? m3$;
990GOTO 1070
1000IF cq=cn(1) OR cq=cn(2) THEN GOTO 1030
1010? bk$; m3$;
1020GOTO 1070
1030INVERSE:? CHR$(8);
1040? " "; cv$(d(cq));
1050NORMAL
1060? m2$;
1070IF cq-INT(cq/6)*6=0 THEN ?:? "  "; :INVERSE:? "R"; x; :NORMAL:? ""; : x=x+1
1075IF x=11 THEN HTAB 1:? "     ";
1080NEXT cq
1084FOR i=1 TO 14
1086VTAB 1:HTAB 27:? "  ";
1088NEXT i
1090RETURN
1100?:? "PRESS "; :INVERSE:? "<RETURN>"; :NORMAL:? " TO CONTINUE."
.1105? " PRESS "; :INVERSE:? " <Q> "; :NORMAL:? " TO QUIT ";
1150INPUT ""; rt$
1160IF rt$="Q" OR rt$="q" THEN GOTO 2000
1190RETURN
2000REM
2040POKE 17059, 0:POKE 17115, 241:POKE  17126,  31:POKE  16953, 95:TEXT:END 3000IF pn<1 OR
pn>4 THEN RETURN
3005IF pn=1 THEN HOME:POKE 17059, 13:POKE 17115,  27:POKE 17126,22:TEXT:RETURN 3010IF pn=2
THEN HOME:POKE  17059,  6:POKE  17115,  31:POKE  17126,  28:TEXT:RETURN  3015IF  pn=3 THEN
HOME:POKE 17059, 7:POKE 17115, 26:POKE  17126,  28:TEXT:RETURN  3020IF pn=4 THEN HOME:POKE
17059, 11:POKE 17115, 24:POKE 17126, 253:TEXT:RETURN
3025GOTO 2040
```

#1 ADAM USERS GROUP - Public Domain Software Exchange

Please continue sending your submissions to the Software Exchange.. Be sure to send ALL correspondence about the Software Exchange to me. I can only any questions if you send a SASE! Will also try to answer phone calls (when I am there) but WILL NOT accept collect calls. I will call you back (I have an answering machine) if you specify to call collect.

STEVE GEORGE - 67 STEVENS AVENUE - OLD BRIDGE, NJ 08857 - (201) 679-6102

A little about the help files on the various disks: SmartWriter HELP files have an "H" designation. CP/M help files have a ".DOC" or ".TXT" extension. These can be read by using the "type" command. You can print them out by hitting the Control key +the P key(this toggles the printer on and off in CP/M). Following is an example: A.> type filename.ext <return> On BASIC disks any file with an "H" designation can be read in SmartWriter. These are usually "README" files.

Rules: For the Software Exchange

1.) Make all checks payable to "STEVE GEORGE". Postal or Money orders are filled immediately. If you send a personal check expect a 3-8 day delay. Stamps are also accepted. Most disks will go out NEXT DAY!! (I hate to wait for software too!).
2.) Questions about programs will be only answered if you send a SASE.

3.) Updated list of programs available send a SASE + $1(to cover costs)
4.) If you have nothing to contribute, a $12 donation is required. I will ONLY supply disk format. This will cover handling, postage, and purchase of disk. If you do not have a disk drive, send a LORAN, M.W. Ruth "PLAIN LABEL", or COLECO DDP + $10 to cover handling and postage (Victory and FastForward DDP's are unreliable). I will also include an updated disk list with your order.
5.) If you have programs to donate, include a README file + $2 to cover postage and handling. I will copy your files & then put the requested disk/DDP on YOUR disk. I will accept DDP only if they are above-mentioned DDPs. You can send me stamps if you want to.
6.) You will not always get the same disk you sent. We use SS,DD disks only.
7.) If you order more than 10 disks take 15% off*20 or more take 20%off
8.) You must be a member of our #1 ADAM USERS'GROUP.

#1 ADAM USERS GROUP - Public Domain Software Exchange - Programs (DP/D)

Disk #1 - CP/M: MBOOT3 - Simple binary file upl/dl, Filter - from ADAM CP/M manual (Chpt 4), FICOPY-file disk copier modifier written in CP/M, can edit anything on disk

Disk #2 - CP/M: Cart Utility, ROMHEX - Dump carts contents in hex/ascii format, Cartcopy -copy carts to disk. Creates.com file. Carts can be run & moved easily., RSTDSK -Tired of typing CTRL-C? Use rstdsk! This disk also has the SOURCE CODE to the cart copier for easy modification.

Disk #3 - BASIC: BALOON - great sprite game, MOD - use w/Packcopy to edit DDP/Disks, 3D Tictactoe, Dueling Cannons - hires game, HELLO -access any disk easily, Brickout - Different than Bonanza version, Lunar Lander-classic text game,LANDER-shape table graphics, STAR TREK-joystick control text adventure, LIFE-graphic population study, BLACKJACK - text,HANOI -graphic game, Paint -has save feature, Eliminator -Hi-res graphic shoot-em'-up, Many other files....

Disk #4 - CP/M: Master Catalog System - Entire cataloguing system for your disks...Access any file from your CP/M disks easily

Disk #5 - BASIC: BACKUP.TAPE - can be modified to backup disks,CARTCOPY - (Not as good as Multicart, but it does work!),TAPEDITOR -Create versions of any sftwre, YAHTZEE, CPMFILTER- to "clean" files from CP/M to BASIC,CROSSFIRE-graphic game, GET EM' -graphic game SMARTDRAW! - w/save feature,COLORTST - test screen colors of monitor Many other files...

Disk #6 - CP/M: LU.COM - Library Utility, SQ.COM - "squeeze" all files 40% with this utility, USQ.COM - "unsqueeze" files that have been squeezed, DUU.COM - Ward Christiansen's CP/M disk utility,LOGALL.COM/.DOC,BUGS.COM,MEMMAP.COM,SORT.COM

Disk #7 - CP/M: Z80 Programmer - Specialized, use w/#8

Disk #8 - CP/M: SCRNCHOP.COM, Z80 Programmers II, Specialized use w/#7

Disk #9 - CP/M: EBASIC.COM - A basic for CP/M..One of the first! Complied NOT interpretive, EBASIC.DOC - Documentation for EBASIC, ERUN.COM - The COMPILER for EBASIC, NSWP.COM/.DOC, EDIT.COM/.DOC -Better than Adam's ED.COM!

Disk #10 - Telecommunications Package: MADAM7-(Use ADAM Modem..Has auto dial/ansfeature.. Better than ADAMLINK II!),MEX(Modem Executive)-Also self-dialing Programmable keys function,Phone book,SCNCRNCH.COM-yet another screen chop Docs for all

Disk #11 - Z80MAC.LBR - (Need LU.COM to extract files) Fully featured assembler, w/docs must for serious programming,SOURCE.LBR-Use w/Z80MAC to ;create Z80 code or change 8080 code to Z80. Can create .ASM files from .COM files! - Includes all docs...also usq.com,lu.com

Disk #12 - CP/M:CP/M ADVENTURE-Classic comes to ADAM includes scr. chop

Disk #13 - BASIC: Personal Finance Planner,CHRUNCHER-similar to Turboload,compresses BASIC programs to run 10 times faster!,COLORCALC DISKINIT,CISPICT-read hires pictures done by other computers*includes examples,HIRESEASEL,MOVEFILES,UTILITYMOD, HI-Q

Disk #14 - BASIC: EVILTEXT, BATTLESHIP, JOINFOUR,CHECKERS, SERPENT, MATHBLS, MENUPRG, SPACE, SCHEDULE,MAYANQUILT All above include doc files

Disk #15 - BASIC: (from NIAD's Exchange..This is a fantastic disk...probably best ADAM PD BASIC disk ever!!),BACKGROUND,BLOCKS,CRAYONS CANON3-music,KBEXAMPLE,NOISEMAKER,

```
                 FDUMP-Fantastic!,BICOLOLR-Background/text color changes,INSTANT,FATFONT-makes
                 ADAM look like PC JR in 40 col mode,SPRITEDIT, CONVERT,FDUMP2USAMAP..much more!
Disk #16 -  BASIC: ROCK w/DOC, HOCKEY w/DOC, GAMESBUS, MASTERMIND,STRONGHOLD w/DOC,
            MINEFIELD w/DOC,TEXTROCKET,METRIC,CONVERTER -converts common metric
            measurements to USA measures and vice versa ·
Disk #17 -  CP/M: MUMPS - a database file system for ADAM VERY complete you need #18 too.
Disk #18 -  CP/M: MUMPS II - use with disk #7 VERY complete!
Disk #19 -  CP/M: SPLIT.COM - Break large data file into small ones,I/OMAP.COM,BASEBALL.
            COM, SECURE.COM, /.COM-link commands,HALLEY-follow the comet,INVENTY.COM,Doc's
Disk #20 -  CP/M: DU-V89 - a more recent version of DUU, TYPTRYT.LBR -a CP/M typing tutor,
            MLOAD24, XCCP.LBR, XCCPUTIL.LBR,MARKET.COM, XRAM -test your RAM disk memory
Disk #21 -  CP/M: SMALL-"C"-a 'C' compiler for the ADAM!, UNIXTOOL.LBR
Disk #22 -  DDTF.COM,.DOC,.LBR, FIND.LBR, HOLD.COM,.DOC, KILL.COM,
            SUPERZAP.LBR, ENCRYPT.LBR, TRIVIA.LBR - Play trivial pursuit
Disk #23 -  CP/M: BISHOW.ASM,.COM, LDIRR.LBR, QK21.PRM, YANCTERM.PRM,CERTIFY.LBR,
            PROBE.LBR, Z80 CHESS, (sue LU.COM or NULU.COM to extract the files)
Disk #24 -  CP/M: COBOL for the ADAM!  Includes ALL one would need to program COBOL on ADAM
Disk #25 -  CP/M: ORGCHOP.COM, FINDBAD.COM, CONT.COM,PRINT.COM,MAKBATCH.COM, DISPLAY.COM,
            SLOAD.COM, CATPAGE.COM, CATLBR.LBR, MCAT -41.LBR
Disk #26 -  CP/M: FORTH for the ADAM!! 1 Full disk with documentation *
            everything one needs to program FORTH on the ADAM!!
Disk #27 -  LOGO: A full disk of logo utilities and games
Disk #28 -  CP/M: A full featured BBS program - RBBS (Remote Bulletin
            Board System) Squeezed 159K!!!
Disk #29 -  Coleco-Never released.......Jeopardy
Disk #30 -  Coleco-Never released.......Troll's Tale
Disk #31 -  Coleco-Never released....Best of Electronic Arts
                 *Hard Hat Mack * Pinball Construction Set
Disk #32 -  Coleco-Never released-Super Sub Roc-90K!! better than cart. version!!
Disk #33 -  Video Hustler..Never released
Disk #34 -  Test Cart..Test system peripherals, printer, DDP, etc..Need
            64K expander, CP/M to run...
Disk #35 -  BASIC- Misc BASIC files from NIAD/AUG Users Group Library
Disk #36 -  BASIC-BBSKit (Updated Bulletin Board Sys. for Adam in BASIC)
Disk #37 -  SmartBasic 2.0 with DOC's
```

**Articles, information, and reviews. LET'S KEEP ADAM MOVING   YOUR HELP IS NEEDED - SEND AS
SOON AS POSSIBLE anything you feel that other members would like see and here about.**

RETURN POSTAGE GUARANTEED

#1 ADAM USERS' GROUP

510 Rhode Island Ave.
Cherry Hill, NJ 08002